AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method, comprising:

    monitoring a node associated with a contended lock; and

    putting a processor that is waiting for~~to acquire~~ the contended lock to become

    available to sleep until an event occurs.

2. (Original) The method of claim 1, wherein the monitoring the node comprises

    monitoring a lock address corresponding to the contended lock by executing a

    monitor instruction to activate the monitoring of the node.

3. (Original) The method of claim 1, further comprises executing a memory wait

    (mwait) instruction to put the processor to sleep until the event occurs.

4. (Original) The method of claim 1, further comprises:

    waking up the processor when the event occurs, wherein the event comprises the

    contended lock becoming available; and

    the processor acquiring the available lock.

5. (Currently Amended) The method of claim 1, wherein the contended lock

    becoming available comprises the processor is next in a queue to acquire the

    contended lock, and the contended lock is released.

6. (Original) The method of claim 1, wherein the putting the processor to sleep

    comprises relinquishing of resources by the processor for other processors to use.

7. (Currently Amended) The method of claim 4, wherein the waking up comprises

    inactivating the monitoring of the node, and the processor using ~~the~~ relinquished

    resources.

8. (Currently Amended) The method of claim ~~7~~6, wherein the relinquishing

    comprises:

relinquishing of a plurality of registers in a register pool;

relinquishing of a plurality of instruction queue entries in an instruction queue;

relinquishing of a plurality of store buffer entries in a store buffer; and

relinquishing of a plurality of re-order buffer entries in a re-order buffer.

9. (Original) A method, comprising:

specifying a monitor address associated with a queue element to monitor the
queue element, wherein the specifying comprises executing a monitor
instruction and a memory wait (mwait) instruction.

10. (Original) The method of claim 9, wherein the queue element corresponds with a
processor to acquire a contended lock.

11. (Original) The method of claim 10, wherein the processor is put to sleep while
waiting for the contended lock using a combination of monitor/mwait.

12. (Original) The method of claim 11, wherein the processor is awakened when an
event occurs, the event comprising the processor is next a queue to acquire the
contended lock, and the contended lock is released.

13. (Currently Amended) A processor, comprising:

an execution unit to execute a monitor instruction and a memory wait (mwait)
instruction to monitor a node associated with a contended lock; and

logic to put a logical processor that is waiting for to acquire the contended lock to
become available to sleep until an event has occurred.

14. (Original) The processor of claim 13, further comprising detection logic to detect
the occurrence of the event, wherein the event comprises a designated event
including the contended lock becoming available.

15.   (Original) The processor of claim 13, wherein the putting the logical processor to sleep comprises relinquishing of resources by the logical processor for other logical processors to use.

16.   (Currently Amended) The processor of claim 13, wherein the logic is further to wake up the logical processor when the event occurs, the waking up comprises inactivating the monitoring of the node, and the logical processing using ~~the~~ relinquished resources.

17.   (Currently Amended) The processor of claim ~~16~~15, wherein the relinquishing comprises:

relinquishing of a plurality of registers in a register pool;

relinquishing of a plurality of instruction queue entries in an instruction queue;

relinquishing of a plurality of store buffer entries in a store buffer; and

relinquishing of a plurality of re-order buffer entries in a re-order buffer.

18.   (Currently Amended) A system comprising:

a storage medium; and

a processor coupled with the storage medium, the processor having

an execution unit to execute a monitor instruction and a memory mwait (mwait)

        instruction to monitor a node associated with a contended lock[[;]], and

logic to put a logical processor that is waiting for ~~to acquire~~ the contended lock to

        become available to sleep until an event has occurred.

19.   (Original) The system of claim 18, further comprising detection logic to detect the occurrence of the event, wherein the event comprises a designated event including the contended lock becoming available.

20. (Original) The system of claim 18, wherein the putting the logical processor to sleep comprises relinquishing of resources by the logical processor for other logical processors to use.

21. (Currently Amended) The system of claim 18, wherein the logic is further to wake up the logical processor when the event occurs, the waking up comprises inactivating the monitoring of the node, and the logical processing using ~~the~~ relinquished resources.

22. (Currently Amended) A machine-readable medium having stored thereon data representing ~~sequences~~ sets of instructions, the ~~sequencing~~ sets of instructions which, when executed by a machine, cause the machine to:

monitor a node associated with a contended lock; and

put a processor that is waiting for ~~to acquire~~ the contended lock to become available to sleep until an event occurs.

23. (Currently Amended) The machine-readable medium of claim 22, wherein the monitoring the node comprises monitoring a lock address corresponding to the contended lock by executing a monitor instruction to activate the monitoring of the node.

24. (Currently Amended) The machine-readable medium of claim 22, wherein the ~~sequences~~ sets of instructions which, when executed by the machine, further cause the machine to execute a memory wait (mwait) instruction to put the processor to sleep until the event occurs.

25.   (Currently Amended) The machine-readable medium of claim 22, wherein the

~~sequences~~ sets of instructions which, when executed by the machine, further

cause the machine to:

wake up the processor when the event occurs, wherein the event comprises the

contended lock becoming available; and

allow the processor to acquire the available lock.

26.   (Currently Amended) The ~~method~~ machine-readable medium of claim 22,

wherein the putting the processor to sleep comprises relinquishing of resources by

the processor for other processors to use.

Claims 27-30 (Canceled)